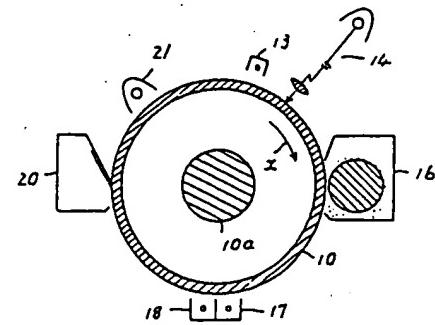


PURPOSE: To obtain an excellent image which has neither fogging nor a decrease in density by providing a heating device nearby an organic photoconductor which is charged electrostatically and positively and holding the temperature of said organic photoconductor at constant temperature or higher.

CONSTITUTION: A photosensitive body 10 incorporates a heater lamp 10a so as to hold its surface temperature at a prescribed temperature or higher. When a power switch is turned on, a lamp 10 is lighted and the photosensitive body 10 is heated to the prescribed temperature. When copying operation is started, the photosensitive body 10 is rotated as shown by an arrow (x) and charged uniformly and positively to enter an exposure process. At this time, the photosensitive body 10 is heated to the prescribed temperature to improve the mobility of positive holes and reduce trapped positive holes, thereby increasing accumulated charges in the photosensitive body 10. Consequently, neither the fogging nor the decrease in density is caused even after repetitive copying operation and a copy image of the same good quality as initial quality is obtained.

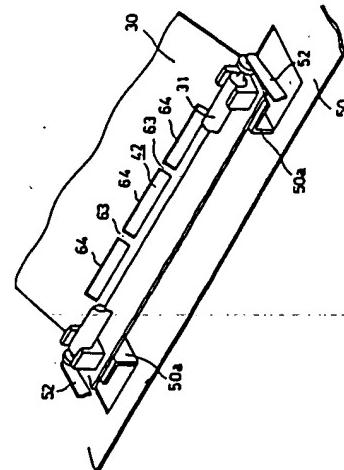


(54) IMAGE FORMING DEVICE

(11) 1-276187 (A) (43) 6.11.1989 (19) JP
(21) Appl. No. 63-104119 (22) 28.4.1988
(71) TOSHIBA CORP(1) (72) HAJIME TAGAWA(4)
(51) Int. Cl'. G03G21/00, G03G15/04

PURPOSE: To prevent a reflection defect and a vibrational sound due to the deformation of a reflector by forming a destaticizing light transmission window in the reflector by slits which have reinforcing ribs.

CONSTITUTION: A destaticizing device is so constituted as to guide part of light from the exposure lamp 31 of an exposure device out of the destaticizing transmission window 41 formed in the reflector 30 and projects it on a photosensitive body. The exposure device is unitized by incorporating respective components in an optical system frame 50. The reflector 30 is fitted on the top surface of this frame 50. The window 41 is composed of the slits 64 which are partitioned discontinuously by the reinforcing ribs 63. Consequently, part of the light from the lamp 31 can be guided to a photosensitive body without spoiling the rigidity of the reflector 30. Then neither the reflection defect nor vibrational sound due to the deformation of the reflector 30 are generated.

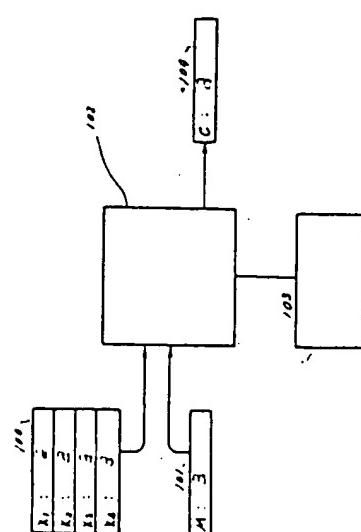


(54) ENCIPHERING SYSTEM

(11) 1-276189 (A) (43) 6.11.1989 (19) JP
(21) Appl. No. 63-103919 (22) 28.4.1988
(71) HITACHI LTD(1) (72) KAZUO TAKARAGI(2)
(51) Int. Cl'. G09C1/00

PURPOSE: To improve the efficiency of enciphering conversion with high-level random property by performing inversion processing for character conversion and data disturbance by combining operation for shifting only specific bits cyclically to the right or left.

CONSTITUTION: A 64-bit normal sentence 101 and key data 100 consisting of 64×4 bits=256 bits are inputted to a microcomputer 102. The computer 102 combines the processing for shifting 32-bit data cyclically to the left or right by 2^n bits ($n \geq 2$ or 16 bits in this case) in the order of an instruction of a program 103 to perform the inversion and conversion processing for data to be enciphered, and outputs a 64-bit enciphered sentence 104 as its result. Consequently, the enciphering conversion with high-level random property is performed efficiently.



⑪ 公開特許公報 (A) 平1-276189

⑫ Int. Cl.
G 09 C 1/00識別記号 庁内整理番号
7368-5B

⑬ 公開 平成1年(1989)11月6日

審査請求 未請求 請求項の数 8 (全9頁)

⑭ 発明の名称 暗号方式

⑮ 特願 昭63-103919

⑯ 出願 昭63(1988)4月28日

| | |
|-------------------------|---|
| ⑰ 発明者 宝木 和夫 | 神奈川県川崎市麻生区王禅寺1099番地 株式会社日立製作所システム開発研究所内 |
| ⑰ 発明者 中川 聰夫 | 茨城県日立市大みか町5丁目2番1号 株式会社日立コントロールシステムズ内 |
| ⑰ 発明者 佐々木 良一 | 神奈川県川崎市麻生区王禅寺1099番地 株式会社日立製作所システム開発研究所内 |
| ⑰ 出願人 株式会社日立製作所 | 東京都千代田区神田駿河台4丁目6番地 |
| ⑰ 出願人 株式会社日立コントロールシステムズ | 茨城県日立市大みか町5丁目2番1号 |
| ⑰ 代理人 弁理士 小川 勝男 | 外1名 |

明細書

1. 発明の名称

暗号方式

2. 特許請求の範囲

1. 換字処理とデータ搅乱のための転置処理を組合せて一定長のデータを暗号化するブロック暗号方式において、上記転置処理は、 2^n ビット ($n = 2$ 以上の整数) だけ右または左に循環シフトするという操作を組合せることにより行なうことを特徴とする暗号方式。

2. 上記換字処理は、ある定数 X として、 X ビットのデータと X ビットの純データとの間で演算を行なうという操作と、 X ビットのデータと該データを右または左に 2^X ビット循環シフトしたものと定数 X との和を 2^X で割った余りをとるという操作を、組合せることにより行なうことを特徴とする第1項の暗号方式。

3. 上記換字処理と転置処理を組み合わせた暗号処理は、暗号化されたデータと、次の暗号化されるべきデータとの間で演算処理を施した後、

該演算結果をさらに暗号化するというフィードバック処理を有することを特徴とする第1項または第2項の暗号方式。

4. 上記転置処理は、32ビットマイクロコンピュータのソフトウェアを用いて、32ビットのデータを、4ビット、または8ビット、または16ビットだけ右または左に循環シフトするという操作を組合せて行なうことを特徴とする第1項～第3項いずれか1項の暗号方式。

5. 上記転置処理は、16ビットマイクロコンピュータのソフトウェアを用いて、16ビットのデータを、4ビット、または8ビット、右または左に循環するという操作を組合せて行なうことを特徴とする第1項～第3項いずれか1項の暗号方式。

6. 上記転置処理は、8ビットマイクロコンピュータのソフトウェアを用いて、8ビットのデータを、4ビット右または左に循環シフトするという操作により行なうことを特徴とする第1項～第3項いずれか1項の暗号方式。

7. 上記換字処理と転置処理とを組み合わせて、メッセージ認証機能を実現するディジタル署名におけるメッセージ認証コード生成のために必要となるメッセージの圧縮文を生成することを特徴とする第1項～第3項いづれか1項の暗号方式。

8. 換字処理とデータ搅乱のための転置処理を組合せて一定長のデータを暗号化するブロック暗号方式において、上記転置処理は、 2^n ビット ($n = 2$ 以上の整数)だけ右または左に循環シフトするという操作を組合せることにより行ない、上記換字処理は、ある定数をXとして、XビットのデータとXビットの鍵データとの間で演算を行なうという操作と、Xビットのデータと該データを右または左に2ビット循環シフトしたものと定数Xとの和を 2^X で割った余りをとるという操作を、組合せることにより行ない、メッセージ認証コードを生成するための圧縮暗号に、上記換字処理と転置処理のすくなくとも一つを用いることにより、メッセージ認証コード

の生成と確認をおこなう機能をICカードに組み入れたことを特徴とする暗号方式。

3. 発明の詳細な説明

(産業上の利用分野)

本発明は、コンピュータのメッセージ等を暗号化する装置に関する。

(従来の技術)

従来の代表的な暗号アルゴリズムとしては、DES (Data Encryption Standard) と FEAL (Fast Encryption Standard) が知られており、DESに関しては例えば、(1) 小山他、「現代暗号理論」、電子通信学会、pp. 41～49、昭和61年9月において、また、FEALに関しては、(2) 清水他、「高速データ暗号アルゴリズムFEAL」、電子通信学会論文誌D、Vol. J70-D、No.7、pp. 1413～1423、1987年7月において、それぞれ詳細に述べられている。

先ず、DESの処理における非線形の計算部分、つまりSボックスといわれる処理について説明す

る(上記(1)のp. 45、図3-2とp.46、図3-3参照)。32ビットのRは、まず、表1に示す拡大型転置表によって置き換えられると共に、一部のビットは重複されて48ビットに拡大されている。

表1 Sボックスの換字表(S_1)

| 列 行 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|--------|----|----|----|---|----|----|----|----|----|----|----|----|----|----|----|----|
| 0 | 14 | 4 | 13 | 1 | 2 | 15 | 11 | 9 | 3 | 10 | 6 | 12 | 5 | 9 | 0 | 7 |
| 1 | 0 | 15 | 7 | 4 | 14 | 2 | 13 | 1 | 10 | 6 | 12 | 11 | 9 | 5 | 3 | 8 |
| 2 | 4 | 1 | 14 | 8 | 13 | 6 | 2 | 11 | 15 | 12 | 9 | 7 | 3 | 10 | 5 | 0 |
| 3 | 15 | 12 | 8 | 2 | 4 | 9 | 1 | 7 | 6 | 11 | 3 | 14 | 10 | 0 | 6 | 13 |

このようにして得られた48ビットのRは、頭から4ビットごとにその後の2ビットを加えた次のような6ビットずつの8組のブロックを形成している。

R₀₁ R₁ R₂ R₃ R₄ R₅,
R₆ R₇ R₈ R₉ R₁₀ R₁₁,

R₁₂ R₁₃ R₁₄ R₁₅ R₁₆ R₁₇,
R₁₈ R₁₉ R₂₀ R₂₁ R₂₂ R₂₃,

この48ビットのR'は、同じく48ビットの鍵Kと排他的論理和の演算を行ない、6ビットずつ8組に分割して、S₁からS₈までの8つのSボックスに入力する。S₁～S₈を選択関数と呼ぶ。これらのSボックスは、6ビットを入力して4ビットを出力する。

例として、表2に一つのSボックスS₁を取り上げてその換字表を示す。

一つのSボックスには、4種類(行番号0, 1, 2, 3)が用意され、この4種類の換字表のどれを用いるかは、入力した6ビットのうち最初と最後のビットを用いて換字表を選ぶ。そして選ばれた換字表にしたがって入力した6ビットの中央の4ビットが換字される。具体的な例として、S₁に対して2進数の入力パターンが011011となっている場合、最初の0と最後の1で表わされ

表2 拡大型転置表 E

| | | | | | |
|----|----|----|----|----|----|
| 32 | 1 | 2 | 3 | 4 | 5 |
| 4 | 5 | 6 | 7 | 8 | 9 |
| 8 | 9 | 10 | 11 | 12 | 13 |
| 12 | 13 | 14 | 15 | 16 | 17 |
| 16 | 17 | 18 | 19 | 20 | 21 |
| 20 | 21 | 22 | 23 | 24 | 25 |
| 24 | 25 | 26 | 27 | 28 | 29 |
| 28 | 29 | 30 | 31 | 32 | 1 |

ている01、つまり行1（2進数01は10進数1であるから）の換字表が選ばれる。次に中央の4ビットのパターン1101（10進数13）で表わされる列13で指定され、この結果行1、列13で指定される値5、つまり0101が出力されて4ビットの換字パターンとなる。DESではこのような処理f(R, K)を用い一段の処理を構成し、これを16段繰り返す。

上記の処理例に見られるように、DESは1ビ

不正使用や盗取等に対する情報セキュリティを確保するため、伝送路上のデータやコンピュータに蓄積されたデータを暗号化することは有効な対策であると考えられる。

昭和52年に、米国商務省標準局が暗号アルゴリズムの標準として制定したDESは、データの暗号化を行う一つの手段である。

ところが、DESはビット単位での処理がたいへん多いため、バイト単位の処理を基調とするマイクロコンピュータのソフトで実現しようとすると、処理に時間がかかり、実用的な速度が得られなかった。

この問題に対し、上記FEALは、1バイト(8ビット)単位の処理を基調とするため、8ビットマイクロコンピュータで実現する場合、DESに比べ數倍以上の高速化を達成することができた。FEALにより、8ビットマイクロコンピュータのソフトを用いてある程度実用的な速度が得られるようになったと考えられる。

しかし、最近のマイクロエレクトロニクスの技

ト単位の処理が基本になっている。

次にFEALの処理における非線形の計算部分、つまり、関数Sを含んでいる部分について説明する（上記(2)のP. 1416、図4及び図5参照）。FEALの非線形部はDESの非線形部に比べ、数学的な記述が簡単である。32ビットデータ α は8ビットのデータ $\alpha^0, \alpha^1, \alpha^2, \alpha^3$ にそれぞれ分割された後、8ビットを単位として、組データと非他の論理和がとられる。その後、所定の関数Sによる処理が実行される。

$$\text{関数 } S : S(x_1 + x_2 + \delta) = \text{Rot}_z(w)$$

$$\text{ただし, } w = (x_1 + x_2 + \delta) \bmod 256$$

$$\delta = 0 \text{ または } 1 \text{ (定数)}$$

この処理f(α, β)を用い、一段の処理を構成し、これが8段繰り返される。上記の処理に見られるように、FEALは8ビット単位の処理が基本になっている。

〔発明が解決しようとする問題点〕

情報処理と通信技術の進歩によるコンピュータ・ネットワークの普及化、大衆化に伴い、データの

術の進歩によって、8ビットマイクロコンピュータよりも16ビットマイクロコンピュータ、さらに、1-8ビットマイクロコンピュータよりも32ビットマイクロコンピュータが使われ出している。近い将来、32ビットマイクロコンピュータが使われる割合がたいへん大きくなると予想されている。32ビットマイクロコンピュータの時代になると、さらに高速の暗号処理が要求されるものと予想される。ところが、32ビットマイクロコンピュータは4バイト基調の処理を行うため、1バイト基調の8ビットマイクロコンピュータ用に設計されたFEALを32ビットマイクロコンピュータで実現しようとすると非効率であった。

そこで、32ビットマイクロコンピュータ向けの4バイト基調の処理を行う暗号アルゴリズムが望まれていた。

〔問題点を解決するための手段〕

上記の問題点を解決するため、次の手段を用いる。

すなわち、32ビットマイクロコンピュータと

メモリからなる暗号変換装置において、
暗号変換されるべきデータの換字変換処理を、
32ビットのデータxとy同士の演算。

$x + y$ 。

つまり、xとyを加算し、 2^{32} で割った余りをとるという処理と、

$\text{Rot}_2(x) + x + \alpha$ 。

つまり、32ビットのデータxを左または右に2ビット循環シフトした後、その結果得られたデータとxと一定値 α を加算した後、 2^{32} で割った余りをとるという処理

を組み合わせることにより実現し、

暗号変換されるデータの転置変換処理を、

$\text{Rot}_4(x)$ 。

第1図において、64ビットの平文101と
 $64\text{ビット} \times 4 = 256\text{ビット}$ の鍵データ100
が32ビットマイクロコンピュータに入力され、
その後、プログラム103内の命令の順に32ビットマイクロコンピュータ102において暗号変換され、その結果として64ビットの暗号文104が出力される。

第2図は、第1図の32ビットマイクロコンピュータ102とプログラム103において実行される暗号変換処理のフローを示している。

201：入力されたデータMは上位32ビット M_1 と下位32ビット M_2 に分割される。

202： M_1 と M_2 のビット対応の排他的論和がとられる。

$\text{WORK}_2 \leftarrow M_1 \oplus M_2$

以下、+は同様の処理を示すものとする。

203： WORK_2 と鍵データ K_1 のモジュロ加算が行われる。

すなわち、32ビットのデータxを左または右へ4ビット循環シフトするという処理と、

$\text{Rot}_8(x)$ 。

すなわち、32ビットのデータxを左または右へ8ビット循環シフトするという処理と、

$\text{Rot}_{16}(x)$ 。

すなわち、32ビットのデータxを左または右へ16ビット循環シフトするという処理を組み合わせることにより実現する。

【作用】

これにより、32ビットマイクロコンピュータを用いて、1回の基本命令で32ビットのデータが換字または転置されるので、暗号変換を高速に行うことができる。

【実施例】

第1図は、本発明の一実施例である。

$x \leftarrow \text{WORK}_2 + K_1$

ここに、 $-x + K_1$ は $-x$ と $-K_1$ の和を -2^{32} で割った余りをとるという、 2^{32} を法としたモジュロ加算を示している。

以下、+は同様の処理を示すものとする。

204： x を左へ2ビット循環シフトした後、そのデータとxと1のモジュロ加算をとる。

$x \leftarrow \text{Rot}_2(x) + x + 1$

以下、 Rot_2 は同様の処理を示すものとする。

105： x を左へ4ビット循環シフトした後、そのデータとxとのモジュロ加算をとる。

$x \leftarrow \text{Rot}_4(x) \oplus x$

以下、 Rot_4 は同様の処理を示すものとする。

206： $\text{WORK}_1 \leftarrow x \oplus M_1$

207： $x \leftarrow x + K_2$

208 : $x \leftarrow \text{Rot}_8(x) + x + 1$ $y \leftarrow x$ 209 : $x \leftarrow \text{Rot}_8(x) \oplus x$

ここに、 $\text{Rot}_8(x)$ は x を左へ 8 ビット循環シフトさせることを示す。

210 : $x \leftarrow x + K_1$ 211 : $x \leftarrow \text{Rot}_8(x) + x + 1$ 212 : $x \leftarrow \text{Rot}_{16}(x) + (x \wedge y)$

ここに、 $\text{Rot}_{16}(x)$ は x を左へ 16 ビット循環シフトすることを示す。また、 $x \wedge y$ は x と y とのビット対応の論理積をとることを示す。

213 : WORK2 $\leftarrow x \oplus \text{WORK2}$ 214 : $x \leftarrow \text{WORK2} + K_4$ 215 : $x \leftarrow \text{Rot}_8(x) + x$ 216 : WORK1 $\leftarrow \text{WORK1} \oplus x$ 217 : WORK2 $\leftarrow \text{WORK2} \oplus \text{WORK1}$

218 : WORK1 を出力データの上位 32 ビット、WORK2 を出力データの下位 32 ビットとして出力する。

以上、第 2 図に示すように関数 $\pi_1 \sim \pi_4$ を定義

すると、本実施例は、

$$C = \pi_1 \cdot \pi_4 \cdot \pi_3 \cdot \pi_2 \cdot \pi_1 (M)$$

というようによ成関数で表すことができる。

関数 $\pi_i \cdot \pi_i (i = 1 \sim 4)$ は、すべて、

$$\pi_i \cdot \pi_i (x) = x$$

というようによ成関数を 2 回繰り返すともとに戻るという性質がある。

したがって、復号関数として、

$$M = \pi_1 \cdot \pi_2 \cdot \pi_3 \cdot \pi_4 \cdot \pi_1 (C)$$

を用いれば、暗号文 C をもとの平文 M に戻すことができる。

実施例の変形例 1

本実施例を 2 回繰り返したものと暗号変換として用いてよい。すなわち、暗号変換を、

$$C = \pi_1 \cdot \pi_2 \cdot \pi_3 \cdot \pi_4 \cdot \pi_1 \\ \cdot \pi_4 \cdot \pi_3 \cdot \pi_2 \cdot \pi_1 (M)$$

$$\text{WORK2} \leftarrow M_1 \oplus M_2$$

としてもよい。

このとき、復号変換の式は

以下、 $+$ は同様の処理を示すものとする。

403 : x と鍵データ K_1 のモジュロ減算が行われる。

$$M = \pi_1 \cdot \pi_2 \cdot \pi_3 \cdot \pi_4 \cdot \pi_1 \\ \cdot \pi_1 \cdot \pi_2 \cdot \pi_3 \cdot \pi_4 \cdot \pi_1 (M)$$

$$x \leftarrow x - K_1$$

である。

同様に、一括に本実施例を n 回繰り返したものと暗号変換としてもよい。

実施例の変形例 2

第 4 図は、本発明の他の実施例である。

401 : 入力されたデータ M は上位 16 ビット M_1 と下位 16 ビット M_2 に分割される。

402 : M_1 と M_2 のビット対応の排他的論理和がとられる。

ここに、 $x - K_1$ は x と K_1 の差を 2^{16} で割った余りをとるという。 2^{16} を法としたモジュロ減算を示している。

以下、 $-$ は同様の処理を示すものとする。

404 : x を左へ 2 ビット循環シフトした後、そのデータと 1 のモジュロ減算を行う。

$$x \leftarrow \text{Rot}(x) - x - 1$$

以下、 Rot_2 は同様の処理を示すものとする。

405 : x を左へ 4 ビット循環シフトした後、そのデータと x の排他的論理和をとる。

$x \leftarrow \text{Rot}_4(x) \oplus x$

以下、 Rot_4 は同様の処理を示すものとする。

406 : WORK1 $\leftarrow x \oplus M_1$,

407 : $x \leftarrow x \leftarrow K_2$

$y \leftarrow x$

408 : $x \leftarrow \text{Rot}_2(x) - x - 1$

409 : $x \leftarrow \text{Rot}_2(x) - (x \wedge y)$

ここに、 $\text{Rot}_2(x)$ は x を左へ8ビット循環シフトすることを示す。また、 $x \wedge y$ は x と y とのビット対応の論理積をとることを示す。

410 : WORK2 $\leftarrow x \oplus \text{WORK2}$

411 : $x \leftarrow \text{WORK2} - K_1$

412 : $x \leftarrow \text{Rot}_2(x) - x - 1$

413 : WORK1 $\leftarrow \text{WORK2} \oplus x$

414 : WORK2 $\leftarrow \text{WORK2} \oplus \text{WORK1}$

415 : WORK1を出力データの上位16ビット、WORK2を出力データの下位16ビットとして出力する。

504 : x を左へ2ビット循環シフトした後、そのデータと $x + 1$ のモジュロ加算を行う。

$x \leftarrow \text{Rot}_2(x) + x + 1$

以下、 Rot_2 は同様の処理を示すものとする。

505 : $x \leftarrow \text{Rot}_2(x) + (x \wedge y)$

ここに、 $\text{Rot}_2(x)$ は x を左へ4ビット循環シフトすることを示す。また、 $x \wedge y$ は x とのビット対応の論理積をとることを示す。

506 : WORK1 $\leftarrow \text{WORK1} \oplus x$

507 : $x \leftarrow \text{WORK1} + K_2$

508 : $x \leftarrow \text{Rot}_2(x) + x + 1$

509 : WORK2 $\leftarrow \text{WORK2} \oplus x$

510 : WORK1 $\leftarrow \text{WORK1} \oplus \text{WORK2}$

511 : WORK1を出力データの上位8ビット、WORK2を出力データの下位8ビットとして出力する。

実施例の変形例4

第6図は本発明の他の一実施例である。

実施例の変形例3

第5図は、本発明の他の実施例である。

501 : 入力されたデータMは上位8ビット M_1 と下位8ビット M_2 に分割される。

502 : M_1 と M_2 のビット対応の排他的論和がとられる。

WORK2 $\leftarrow M_1 \oplus M_2$

以下、+は同様の処理を示すものとする。

503 : x と鍵データ K_1 のモジュロ加算が行われる。

$x \leftarrow \text{WORK2} + K_1$

$y \leftarrow x$

ここに、 $x + K_1$ は x と K_1 の差を 2^8 で割った余りをとるという、 2^8 を法としたモジュロ加算を示している。

以下、+は同様の処理を示すものとする。

(1) 認証を行うメッセージ62を鍵データとして、任意の初期値61を本発明によるアルゴリズム6-3を用いて暗号化する。

(2) 暗号結果64を、(1)において用いたメッセージの続きのデータにより再び暗号化し、メッセージの終わりまでこの操作を繰り返す。

(3) 最終的な暗号結果をメッセージ認証コード65として出力する。

実施例の変形例5

第7図は本発明の他の実施例である。本ICカードは、第7項記載の方式によりメッセージの認証コードを生成する。

(1) メッセージの認証を行うために必要な初期値76をI/O74を通して、ICカード71内のマイクロコンピュータ72に送信する。

(2) 認証を行うメッセージ77を(1)と同様にマイクロコンピュータ72に順次送信し、マイクロコンピュータ72は、メモリ73に記憶されている暗号ソフト75により認証コード78を生成する。

〔効果〕

本実施例は、第3図に示すような換字、転置の繰返しを行っている。

つまり、第2図に示す実施例。

(203, 204), (207, 208),

(210, 211), (214, 215) の処理は、

$$\begin{aligned} x &\leftarrow x + K_i \\ x &\leftarrow \text{Rot}_z(x) + (x) + 1 \end{aligned}$$

の形となっており、これは、それぞれ、32ビットのデータを4ビットずつのブロックに分割したとき、各ブロック単位の換字処理を、上記2回のデータ変換により8ブロック分一斉に行っていると見ることができる。

ここに、4ビットのブロックデータ

$$\begin{aligned} A &= (a_1, a_2, a_3, a_4), \text{ただし}, \\ a_i &= 1 \text{ or } 0 \quad (i = 1 \sim 4) \end{aligned}$$

が、

の処理を行っており、これらは、それぞれ、

(1) 4ビット左循環シフトを行うという転置を行った後、さらに換字を行うという処理。

(2) 8ビット左循環シフトを行うという転置を行った後、さらに換字を行うという処理。

(2) 16ビット左循環シフトを行うという処理を示している。

第3図から明らかのように、最初の32ビットのデータのうち、いかなるビットの変化も最後の32ビットのデータすべてに影響を与えることが分かる。

これにより、本実施例は、高度なランダム性を持つ暗号変換を効率良く行うという効果が得られることが分かる。

4. 図面の簡単な説明

第1図は、本発明を実施する暗号変換装置の一実施例、第2図は、第1図における暗号変換の詳細を示すフローチャート、第3図は、本発明の実施例が効率的に換字変換、転置変換を繰り返していることを示す説明図、第4図は、16ビットマ

$B = (b_1, b_2, b_3, b_4)$ 、ただし、

$b_i = 1 \text{ or } 0 \quad (i = 1 \sim 4)$

に換字変換されるということは、

プール代数の演算 f_1, f_2, f_3, f_4 が存在して、

$$\begin{aligned} b_1 &= f_1(a_1, a_2, a_3, a_4) \\ b_2 &= f_2(a_1, a_2, a_3, a_4) \\ b_3 &= f_3(a_1, a_2, a_3, a_4) \\ b_4 &= f_4(a_1, a_2, a_3, a_4) \end{aligned}$$

となることを示す。

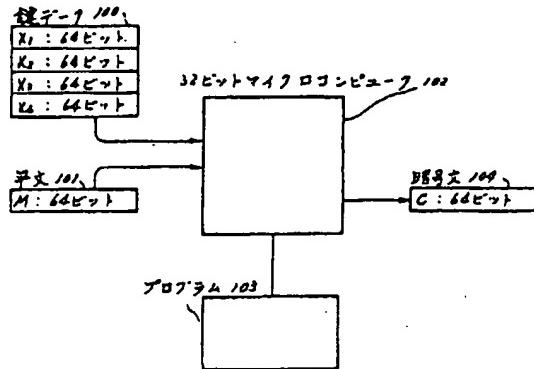
また、第2図の205, 209, 212はそれぞれ、

- (1) $x \leftarrow \text{Rot}_4(x) \oplus x$
- (2) $x \leftarrow \text{Rot}_8(x) \oplus x$
- (3) $x \leftarrow \text{Rot}_{16}(x) + (x \wedge y)$

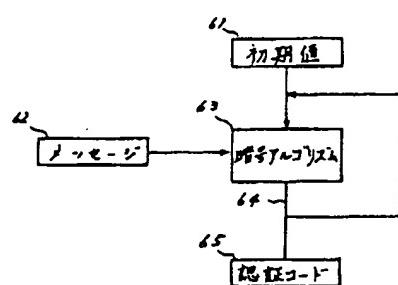
イクロコンピュータを用いた場合の暗号変換の詳細を示すフローチャート、第5図は、8ビットマイクロコンピュータを用いた場合の暗号変換の詳細を示すフローチャート、第6図は、本発明による暗号アルゴリズムを用いてメッセージ認証コードを生成する方法を示すフローチャート、第7図は、本発明による暗号アルゴリズムを用いてメッセージ認証コードを生成するICカードの構成図である。

代理人弁理士 小川勝男

第一回

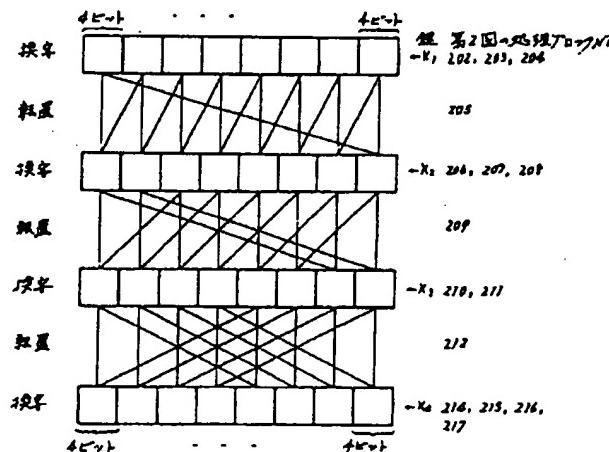


第 6 四

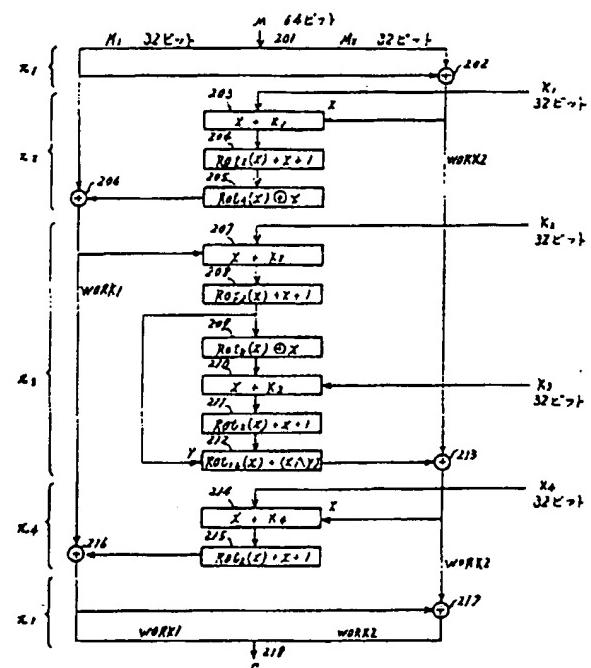


第 3 回

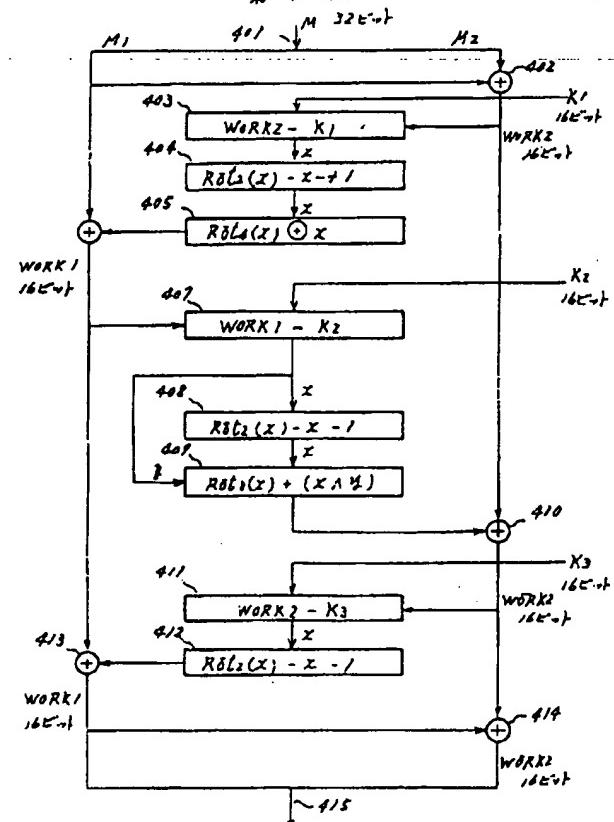
M₁ (上位32ビット) または *M₂* (下位32ビット) の変換用関数



五 2

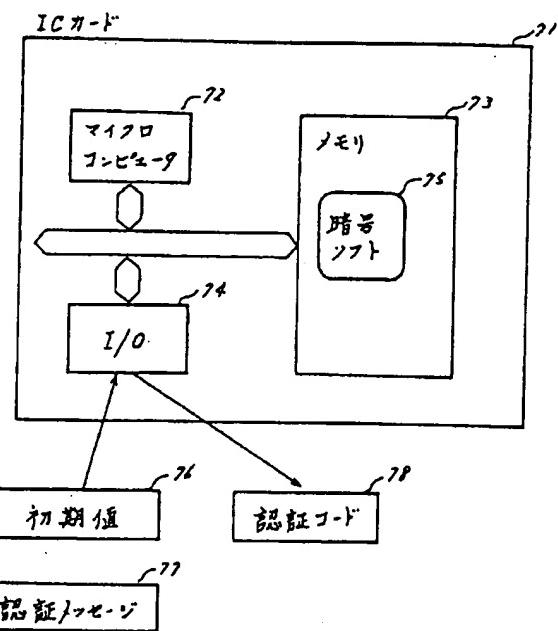
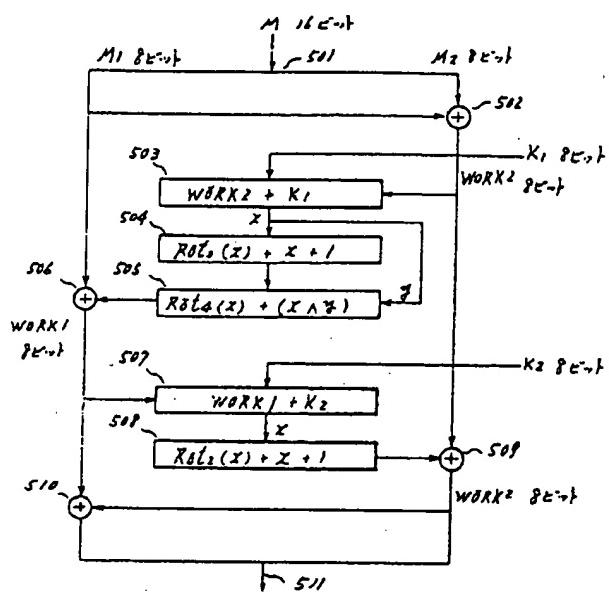


第4回



第7図

第5図



【公報種別】特許法第17条の2の規定による補正の掲載

【部門区分】第6部門第2区分

【発行日】平成8年(1996)7月12日

【公開番号】特開平1-276189

【公開日】平成1年(1989)11月6日

【年通号数】公開特許公報1-2762

【出願番号】特願昭63-103919

【国際特許分類第6版】

G09C 1/00 9364-5L

…補正の内容…

1. 明細書の発明の名称の欄の記載を「暗号化方法及び暗号の復号化方法」に訂正する。

2. 明細書の特許請求の範囲の欄の記載を別紙のとおりに補正する。

3. 明細書の発明の詳細な説明の欄について以下の補正を行う。

(1) 明細書第16頁第13行「本実施例を」を「上記実施例における実施例
X:からX:までにあたる処理を」に訂正する。

(2) 明細書第17頁第1~2行の式を次のとおりに訂正する。

「 G = X₁ X₂ X₃ X₄ X₅ X₆ X₇ X₈ X₉ X₁₀ X₁₁ X₁₂ X₁₃ X₁₄ X₁₅ X₁₆ X₁₇ X₁₈ X₁₉ X₂₀ X₂₁ X₂₂ X₂₃ X₂₄ X₂₅ X₂₆ X₂₇ X₂₈ X₂₉ X₃₀ X₃₁ X₃₂ X₃₃ X₃₄ X₃₅ X₃₆ X₃₇ X₃₈ X₃₉ X₄₀ X₄₁ X₄₂ X₄₃ X₄₄ X₄₅ X₄₆ X₄₇ X₄₈ X₄₉ X₅₀ X₅₁ X₅₂ X₅₃ X₅₄ X₅₅ X₅₆ X₅₇ X₅₈ X₅₉ X₆₀ X₆₁ X₆₂ X₆₃ X₆₄ X₆₅ X₆₆ X₆₇ X₆₈ X₆₉ X₇₀ X₇₁ X₇₂ X₇₃ X₇₄ X₇₅ X₇₆ X₇₇ X₇₈ X₇₉ X₈₀ X₈₁ X₈₂ X₈₃ X₈₄ X₈₅ X₈₆ X₈₇ X₈₈ X₈₉ X₉₀ X₉₁ X₉₂ X₉₃ X₉₄ X₉₅ X₉₆ X₉₇ X₉₈ X₉₉ X₁₀₀ X₁₀₁ X₁₀₂ X₁₀₃ X₁₀₄ X₁₀₅ X₁₀₆ X₁₀₇ X₁₀₈ X₁₀₉ X₁₁₀ X₁₁₁ X₁₁₂ X₁₁₃ X₁₁₄ X₁₁₅ X₁₁₆ X₁₁₇ X₁₁₈ X₁₁₉ X₁₂₀ X₁₂₁ X₁₂₂ X₁₂₃ X₁₂₄ X₁₂₅ X₁₂₆ X₁₂₇ X₁₂₈ X₁₂₉ X₁₃₀ X₁₃₁ X₁₃₂ X₁₃₃ X₁₃₄ X₁₃₅ X₁₃₆ X₁₃₇ X₁₃₈ X₁₃₉ X₁₄₀ X₁₄₁ X₁₄₂ X₁₄₃ X₁₄₄ X₁₄₅ X₁₄₆ X₁₄₇ X₁₄₈ X₁₄₉ X₁₅₀ X₁₅₁ X₁₅₂ X₁₅₃ X₁₅₄ X₁₅₅ X₁₅₆ X₁₅₇ X₁₅₈ X₁₅₉ X₁₆₀ X₁₆₁ X₁₆₂ X₁₆₃ X₁₆₄ X₁₆₅ X₁₆₆ X₁₆₇ X₁₆₈ X₁₆₉ X₁₇₀ X₁₇₁ X₁₇₂ X₁₇₃ X₁₇₄ X₁₇₅ X₁₇₆ X₁₇₇ X₁₇₈ X₁₇₉ X₁₈₀ X₁₈₁ X₁₈₂ X₁₈₃ X₁₈₄ X₁₈₅ X₁₈₆ X₁₈₇ X₁₈₈ X₁₈₉ X₁₉₀ X₁₉₁ X₁₉₂ X₁₉₃ X₁₉₄ X₁₉₅ X₁₉₆ X₁₉₇ X₁₉₈ X₁₉₉ X₂₀₀ X₂₀₁ X₂₀₂ X₂₀₃ X₂₀₄ X₂₀₅ X₂₀₆ X₂₀₇ X₂₀₈ X₂₀₉ X₂₁₀ X₂₁₁ X₂₁₂ X₂₁₃ X₂₁₄ X₂₁₅ X₂₁₆ X₂₁₇ X₂₁₈ X₂₁₉ X₂₂₀ X₂₂₁ X₂₂₂ X₂₂₃ X₂₂₄ X₂₂₅ X₂₂₆ X₂₂₇ X₂₂₈ X₂₂₉ X₂₃₀ X₂₃₁ X₂₃₂ X₂₃₃ X₂₃₄ X₂₃₅ X₂₃₆ X₂₃₇ X₂₃₈ X₂₃₉ X₂₄₀ X₂₄₁ X₂₄₂ X₂₄₃ X₂₄₄ X₂₄₅ X₂₄₆ X₂₄₇ X₂₄₈ X₂₄₉ X₂₅₀ X₂₅₁ X₂₅₂ X₂₅₃ X₂₅₄ X₂₅₅ X₂₅₆ X₂₅₇ X₂₅₈ X₂₅₉ X₂₆₀ X₂₆₁ X₂₆₂ X₂₆₃ X₂₆₄ X₂₆₅ X₂₆₆ X₂₆₇ X₂₆₈ X₂₆₉ X₂₇₀ X₂₇₁ X₂₇₂ X₂₇₃ X₂₇₄ X₂₇₅ X₂₇₆ X₂₇₇ X₂₇₈ X₂₇₉ X₂₈₀ X₂₈₁ X₂₈₂ X₂₈₃ X₂₈₄ X₂₈₅ X₂₈₆ X₂₈₇ X₂₈₈ X₂₈₉ X₂₉₀ X₂₉₁ X₂₉₂ X₂₉₃ X₂₉₄ X₂₉₅ X₂₉₆ X₂₉₇ X₂₉₈ X₂₉₉ X₃₀₀ X₃₀₁ X₃₀₂ X₃₀₃ X₃₀₄ X₃₀₅ X₃₀₆ X₃₀₇ X₃₀₈ X₃₀₉ X₃₁₀ X₃₁₁ X₃₁₂ X₃₁₃ X₃₁₄ X₃₁₅ X₃₁₆ X₃₁₇ X₃₁₈ X₃₁₉ X₃₂₀ X₃₂₁ X₃₂₂ X₃₂₃ X₃₂₄ X₃₂₅ X₃₂₆ X₃₂₇ X₃₂₈ X₃₂₉ X₃₃₀ X₃₃₁ X₃₃₂ X₃₃₃ X₃₃₄ X₃₃₅ X₃₃₆ X₃₃₇ X₃₃₈ X₃₃₉ X₃₄₀ X₃₄₁ X₃₄₂ X₃₄₃ X₃₄₄ X₃₄₅ X₃₄₆ X₃₄₇ X₃₄₈ X₃₄₉ X₃₅₀ X₃₅₁ X₃₅₂ X₃₅₃ X₃₅₄ X₃₅₅ X₃₅₆ X₃₅₇ X₃₅₈ X₃₅₉ X₃₆₀ X₃₆₁ X₃₆₂ X₃₆₃ X₃₆₄ X₃₆₅ X₃₆₆ X₃₆₇ X₃₆₈ X₃₆₉ X₃₇₀ X₃₇₁ X₃₇₂ X₃₇₃ X₃₇₄ X₃₇₅ X₃₇₆ X₃₇₇ X₃₇₈ X₃₇₉ X₃₈₀ X₃₈₁ X₃₈₂ X₃₈₃ X₃₈₄ X₃₈₅ X₃₈₆ X₃₈₇ X₃₈₈ X₃₈₉ X₃₉₀ X₃₉₁ X₃₉₂ X₃₉₃ X₃₉₄ X₃₉₅ X₃₉₆ X₃₉₇ X₃₉₈ X₃₉₉ X₄₀₀ X₄₀₁ X₄₀₂ X₄₀₃ X₄₀₄ X₄₀₅ X₄₀₆ X₄₀₇ X₄₀₈ X₄₀₉ X₄₁₀ X₄₁₁ X₄₁₂ X₄₁₃ X₄₁₄ X₄₁₅ X₄₁₆ X₄₁₇ X₄₁₈ X₄₁₉ X₄₂₀ X₄₂₁ X₄₂₂ X₄₂₃ X₄₂₄ X₄₂₅ X₄₂₆ X₄₂₇ X₄₂₈ X₄₂₉ X₄₃₀ X₄₃₁ X₄₃₂ X₄₃₃ X₄₃₄ X₄₃₅ X₄₃₆ X₄₃₇ X₄₃₈ X₄₃₉ X₄₄₀ X₄₄₁ X₄₄₂ X₄₄₃ X₄₄₄ X₄₄₅ X₄₄₆ X₄₄₇ X₄₄₈ X₄₄₉ X₄₅₀ X₄₅₁ X₄₅₂ X₄₅₃ X₄₅₄ X₄₅₅ X₄₅₆ X₄₅₇ X₄₅₈ X₄₅₉ X₄₆₀ X₄₆₁ X₄₆₂ X₄₆₃ X₄₆₄ X₄₆₅ X₄₆₆ X₄₆₇ X₄₆₈ X₄₆₉ X₄₇₀ X₄₇₁ X₄₇₂ X₄₇₃ X₄₇₄ X₄₇₅ X₄₇₆ X₄₇₇ X₄₇₈ X₄₇₉ X₄₈₀ X₄₈₁ X₄₈₂ X₄₈₃ X₄₈₄ X₄₈₅ X₄₈₆ X₄₈₇ X₄₈₈ X₄₈₉ X₄₉₀ X₄₉₁ X₄₉₂ X₄₉₃ X₄₉₄ X₄₉₅ X₄₉₆ X₄₉₇ X₄₉₈ X₄₉₉ X₅₀₀ X₅₀₁ X₅₀₂ X₅₀₃ X₅₀₄ X₅₀₅ X₅₀₆ X₅₀₇ X₅₀₈ X₅₀₉ X₅₁₀ X₅₁₁ X₅₁₂ X₅₁₃ X₅₁₄ X₅₁₅ X₅₁₆ X₅₁₇ X₅₁₈ X₅₁₉ X₅₂₀ X₅₂₁ X₅₂₂ X₅₂₃ X₅₂₄ X₅₂₅ X₅₂₆ X₅₂₇ X₅₂₈ X₅₂₉ X₅₃₀ X₅₃₁ X₅₃₂ X₅₃₃ X₅₃₄ X₅₃₅ X₅₃₆ X₅₃₇ X₅₃₈ X₅₃₉ X₅₄₀ X₅₄₁ X₅₄₂ X₅₄₃ X₅₄₄ X₅₄₅ X₅₄₆ X₅₄₇ X₅₄₈ X₅₄₉ X₅₅₀ X₅₅₁ X₅₅₂ X₅₅₃ X₅₅₄ X₅₅₅ X₅₅₆ X₅₅₇ X₅₅₈ X₅₅₉ X₅₆₀ X₅₆₁ X₅₆₂ X₅₆₃ X₅₆₄ X₅₆₅ X₅₆₆ X₅₆₇ X₅₆₈ X₅₆₉ X₅₇₀ X₅₇₁ X₅₇₂ X₅₇₃ X₅₇₄ X₅₇₅ X₅₇₆ X₅₇₇ X₅₇₈ X₅₇₉ X₅₈₀ X₅₈₁ X₅₈₂ X₅₈₃ X₅₈₄ X₅₈₅ X₅₈₆ X₅₈₇ X₅₈₈ X₅₈₉ X₅₉₀ X₅₉₁ X₅₉₂ X₅₉₃ X₅₉₄ X₅₉₅ X₅₉₆ X₅₉₇ X₅₉₈ X₅₉₉ X₆₀₀ X₆₀₁ X₆₀₂ X₆₀₃ X₆₀₄ X₆₀₅ X₆₀₆ X₆₀₇ X₆₀₈ X₆₀₉ X₆₁₀ X₆₁₁ X₆₁₂ X₆₁₃ X₆₁₄ X₆₁₅ X₆₁₆ X₆₁₇ X₆₁₈ X₆₁₉ X₆₂₀ X₆₂₁ X₆₂₂ X₆₂₃ X₆₂₄ X₆₂₅ X₆₂₆ X₆₂₇ X₆₂₈ X₆₂₉ X₆₃₀ X₆₃₁ X₆₃₂ X₆₃₃ X₆₃₄ X₆₃₅ X₆₃₆ X₆₃₇ X₆₃₈ X₆₃₉ X₆₄₀ X₆₄₁ X₆₄₂ X₆₄₃ X₆₄₄ X₆₄₅ X₆₄₆ X₆₄₇ X₆₄₈ X₆₄₉ X₆₅₀ X₆₅₁ X₆₅₂ X₆₅₃ X₆₅₄ X₆₅₅ X₆₅₆ X₆₅₇ X₆₅₈ X₆₅₉ X₆₆₀ X₆₆₁ X₆₆₂ X₆₆₃ X₆₆₄ X₆₆₅ X₆₆₆ X₆₆₇ X₆₆₈ X₆₆₉ X₆₇₀ X₆₇₁ X₆₇₂ X₆₇₃ X₆₇₄ X₆₇₅ X₆₇₆ X₆₇₇ X₆₇₈ X₆₇₉ X₆₈₀ X₆₈₁ X₆₈₂ X₆₈₃ X₆₈₄ X₆₈₅ X₆₈₆ X₆₈₇ X₆₈₈ X₆₈₉ X₆₉₀ X₆₉₁ X₆₉₂ X₆₉₃ X₆₉₄ X₆₉₅ X₆₉₆ X₆₉₇ X₆₉₈ X₆₉₉ X₇₀₀ X₇₀₁ X₇₀₂ X₇₀₃ X₇₀₄ X₇₀₅ X₇₀₆ X₇₀₇ X₇₀₈ X₇₀₉ X₇₁₀ X₇₁₁ X₇₁₂ X₇₁₃ X₇₁₄ X₇₁₅ X₇₁₆ X₇₁₇ X₇₁₈ X₇₁₉ X₇₂₀ X₇₂₁ X₇₂₂ X₇₂₃ X₇₂₄ X₇₂₅ X₇₂₆ X₇₂₇ X₇₂₈ X₇₂₉ X₇₃₀ X₇₃₁ X₇₃₂ X₇₃₃ X₇₃₄ X₇₃₅ X₇₃₆ X₇₃₇ X₇₃₈ X₇₃₉ X₇₄₀ X₇₄₁ X₇₄₂ X₇₄₃ X₇₄₄ X₇₄₅ X₇₄₆ X₇₄₇ X₇₄₈ X₇₄₉ X₇₅₀ X₇₅₁ X₇₅₂ X₇₅₃ X₇₅₄ X₇₅₅ X₇₅₆ X₇₅₇ X₇₅₈ X₇₅₉ X₇₆₀ X₇₆₁ X₇₆₂ X₇₆₃ X₇₆₄ X₇₆₅ X₇₆₆ X₇₆₇ X₇₆₈ X₇₆₉ X₇₇₀ X₇₇₁ X₇₇₂ X₇₇₃ X₇₇₄ X₇₇₅ X₇₇₆ X₇₇₇ X₇₇₈ X₇₇₉ X₇₈₀ X₇₈₁ X₇₈₂ X₇₈₃ X₇₈₄ X₇₈₅ X₇₈₆ X₇₈₇ X₇₈₈ X₇₈₉ X₇₉₀ X₇₉₁ X₇₉₂ X₇₉₃ X₇₉₄ X₇₉₅ X₇₉₆ X₇₉₇ X₇₉₈ X₇₉₉ X₈₀₀ X₈₀₁ X₈₀₂ X₈₀₃ X₈₀₄ X₈₀₅ X₈₀₆ X₈₀₇ X₈₀₈ X₈₀₉ X₈₁₀ X₈₁₁ X₈₁₂ X₈₁₃ X₈₁₄ X₈₁₅ X₈₁₆ X₈₁₇ X₈₁₈ X₈₁₉ X₈₂₀ X₈₂₁ X₈₂₂ X₈₂₃ X₈₂₄ X₈₂₅ X₈₂₆ X₈₂₇ X₈₂₈ X₈₂₉ X₈₃₀ X₈₃₁ X₈₃₂ X₈₃₃ X₈₃₄ X₈₃₅ X₈₃₆ X₈₃₇ X₈₃₈ X₈₃₉ X₈₄₀ X₈₄₁ X₈₄₂ X₈₄₃ X₈₄₄ X₈₄₅ X₈₄₆ X₈₄₇ X₈₄₈ X₈₄₉ X₈₅₀ X₈₅₁ X₈₅₂ X₈₅₃ X₈₅₄ X₈₅₅ X₈₅₆ X₈₅₇ X₈₅₈ X₈₅₉ X₈₆₀ X₈₆₁ X₈₆₂ X₈₆₃ X₈₆₄ X₈₆₅ X₈₆₆ X₈₆₇ X₈₆₈ X₈₆₉ X₈₇₀ X₈₇₁ X₈₇₂ X₈₇₃ X₈₇₄ X₈₇₅ X₈₇₆ X₈₇₇ X₈₇₈ X₈₇₉ X₈₈₀ X₈₈₁ X₈₈₂ X₈₈₃ X₈₈₄ X₈₈₅ X₈₈₆ X₈₈₇ X₈₈₈ X₈₈₉ X₈₉₀ X₈₉₁ X₈₉₂ X₈₉₃ X₈₉₄ X₈₉₅ X₈₉₆ X₈₉₇ X₈₉₈ X₈₉₉ X₉₀₀ X₉₀₁ X₉₀₂ X₉₀₃ X₉₀₄ X₉₀₅ X₉₀₆ X₉₀₇ X₉₀₈ X₉₀₉ X₉₁₀ X₉₁₁ X₉₁₂ X₉₁₃ X₉₁₄ X₉₁₅ X₉₁₆ X₉₁₇ X₉₁₈ X₉₁₉ X₉₂₀ X₉₂₁ X₉₂₂ X₉₂₃ X₉₂₄ X₉₂₅ X₉₂₆ X₉₂₇ X₉₂₈ X₉₂₉ X₉₃₀ X₉₃₁ X₉₃₂ X₉₃₃ X₉₃₄ X₉₃₅ X₉₃₆ X₉₃₇ X₉₃₈ X₉₃₉ X₉₄₀ X₉₄₁ X₉₄₂ X₉₄₃ X₉₄₄ X₉₄₅ X₉₄₆ X₉₄₇ X₉₄₈ X₉₄₉ X₉₅₀ X₉₅₁ X₉₅₂ X₉₅₃ X₉₅₄ X₉₅₅ X₉₅₆ X₉₅₇ X₉₅₈ X₉₅₉ X₉₆₀ X₉₆₁ X₉₆₂ X₉₆₃ X₉₆₄ X₉₆₅ X₉₆₆ X₉₆₇ X₉₆₈ X₉₆₉ X₉₇₀ X₉₇₁ X₉₇₂ X₉₇₃ X₉₇₄ X₉₇₅ X₉₇₆ X₉₇₇ X₉₇₈ X₉₇₉ X₉₈₀ X₉₈₁ X₉₈₂ X₉₈₃ X₉₈₄ X₉₈₅ X₉₈₆ X₉₈₇ X₉₈₈ X₉₈₉ X₉₉₀ X₉₉₁ X₉₉₂ X₉₉₃ X₉₉₄ X₉₉₅ X₉₉₆ X₉₉₇ X₉₉₈ X₉₉₉ X₁₀₀₀ X₁₀₀₁ X₁₀₀₂ X₁₀₀₃ X₁₀₀₄ X₁₀₀₅ X₁₀₀₆ X₁₀₀₇ X₁₀₀₈ X₁₀₀₉ X₁₀₁₀ X₁₀₁₁ X₁₀₁₂ X₁₀₁₃ X₁₀₁₄ X₁₀₁₅ X₁₀₁₆ X₁₀₁₇ X₁₀₁₈ X₁₀₁₉ X₁₀₂₀ X₁₀₂₁ X₁₀₂₂ X₁₀₂₃ X₁₀₂₄ X₁₀₂₅ X₁₀₂₆ X₁₀₂₇ X₁₀₂₈ X₁₀₂₉ X₁₀₃₀ X₁₀₃₁ X₁₀₃₂ X₁₀₃₃ X₁₀₃₄ X₁₀₃₅ X₁₀₃₆ X₁₀₃₇ X₁₀₃₈ X₁₀₃₉ X₁₀₄₀ X₁₀₄₁ X₁₀₄₂ X₁₀₄₃ X₁₀₄₄ X₁₀₄₅ X₁₀₄₆ X₁₀₄₇ X₁₀₄₈ X₁₀₄₉ X₁₀₅₀ X₁₀₅₁ X₁₀₅₂ X₁₀₅₃ X₁₀₅₄ X₁₀₅₅ X₁₀₅₆ X₁₀₅₇ X₁₀₅₈ X_{1059</sub}

別紙

特許請求の範囲

1. 暗号化対象データに対して所定ビット長のブロックごとに暗号化処理を行う暗号化方法において、該データをそれぞれ用いてデータへ換字処理を施すこと、およびデータ混乱のためデータに転置処理を施すことを交互に実施し、上記転置処理は、 2^n ビット（ $n = 2$ 以上の整数）だけ右もしくは左に処理中のデータを循環シフトする操作を含む所定の暗号化方法で作成された暗号を復号する復号化方法であり、上記所定の暗号化方法の処理ステップ全体をそれぞれ同一の回数実行を2度行えばデータが元に戻る回数実行の復号版用の積み重ねとしたとき、上記作成された暗号に対し、上記複数段階の復号版用を上記暗号化方法とは逆の順序で実行することを特徴とする暗号の復号化方法。
2. 上記転置処理における循環シフトの深さは換字処理と転置処理の交互実施ごとに 2^0 ビット、 2^1 ビット、 \dots 、と順次倍増することを特徴とする特許請求の範囲第1項記載の暗号化方法。
3. 上記換字処理は、3, 2ビットのデータを4ビット、または8ビット、または16ビットだけ右もしくは左に循環シフトする操作を含むことを特徴とする特許請求の範囲第1項記載の暗号化方法。
4. 上記換字処理と転置処理を組み合せた暗号化処理は、暗号化されたデータと、次の暗号化されるべきデータ戸の合いで複数処理を施した後、該複数処理結果をさらに暗号化するというフィードバック処理を有することを特徴とする特許請求の範囲第1項記載の暗号化方法。
5. 該データをそれぞれ用いてデータへ換字処理を施すこと、およびデータ混乱のためデータに転置処理を施すことを交互に実施し、上記転置処理は、 2^n ビット（ $n = 2$ 以上の整数）だけ右もしくは左に処理中のデータを循環シフトする操作を含む所定の暗号化方法用い、デジタル署名により認証すべきメッセージを順次切り出して上記該データとし、所定の初期値を上記所定の暗号化方法で順次暗号化してメッセージ承認コードとするメッセージの認証方法。
6. 該データをそれぞれ用いてデータへ換字処理を施すこと、およびデータ混乱のためデータに転置処理を施すことを交互に実施し、上記転置処理は、 2^n ビット（ $n = 2$ 以上の整数）だけ右もしくは左に処理中のデータを循環シフトする操作を含む所定の暗号化アルゴリズムをICカードの記憶領域に記憶し、上記ICカードでは認証すべきメッセージと初期データとが入力すると、該メッセージを順次切り出して上記該データとし、上記初期データに対し上記所定の

暗号化アルゴリズムによる暗号化処理を順次施してメッセージ承認コードを発生することを特徴とするメッセージの認証方法。

7. 該データをそれぞれ用いてデータへ換字処理を施すこと、およびデータ混乱のためデータに転置処理を施すことを交互に実施し、上記転置処理は、 2^n ビット（ $n = 2$ 以上の整数）だけ右もしくは左に処理中のデータを循環シフトする操作を含む所定の暗号化方法で作成された暗号を復号する復号化方法であり、上記所定の暗号化方法の処理ステップ全体をそれぞれ同一の回数実行を2度行えばデータが元に戻る回数実行の復号版用の積み重ねとしたとき、上記作成された暗号に対し、上記複数段階の復号版用を上記暗号化方法とは逆の順序で実行することを特徴とする暗号の復号化方法。